This version of the deck has notes like this in Times New Roman on some slides to explain what's going on for those who couldn't see it live!

See the accompanying blog post at https://ihaque.org/posts/2019/03/25/three-principles-for-ai-ml/

The organizers asked me to give a talk that was not "about some [particular AI/ML] method, but about **what principles we ought to be observing, or at least aware of**" when planning AI projects in drug discovery.

Thanks, I Hate It!

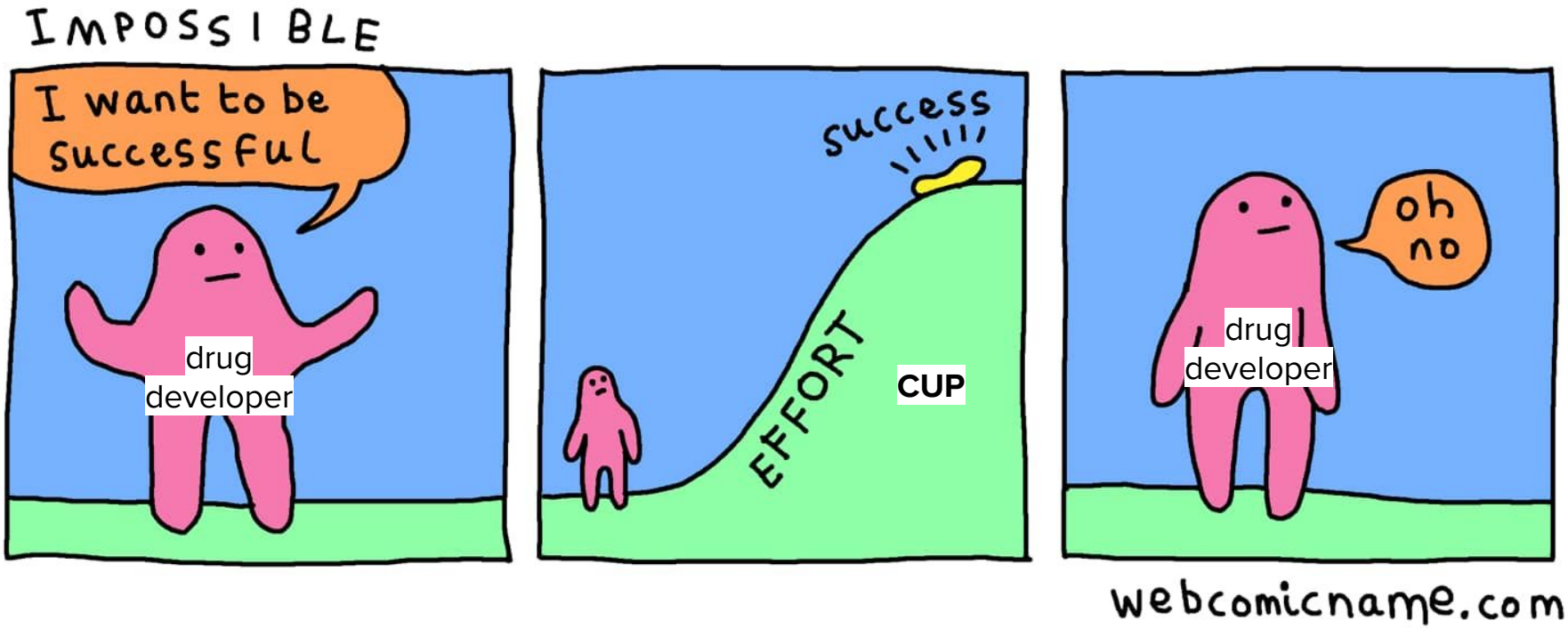
Why your biological machine learning model probably won't work (and what to do about it)

Imran S. Haque, PhD

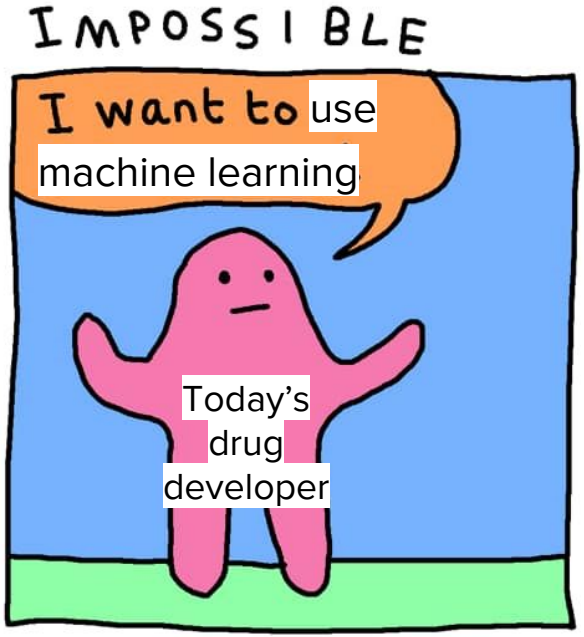
Twitter: @imranshaque
https://ihaque.org
ish@ihaque.org

7 Mar 2019
OpenEye CUP XIX
Santa Fe, NM

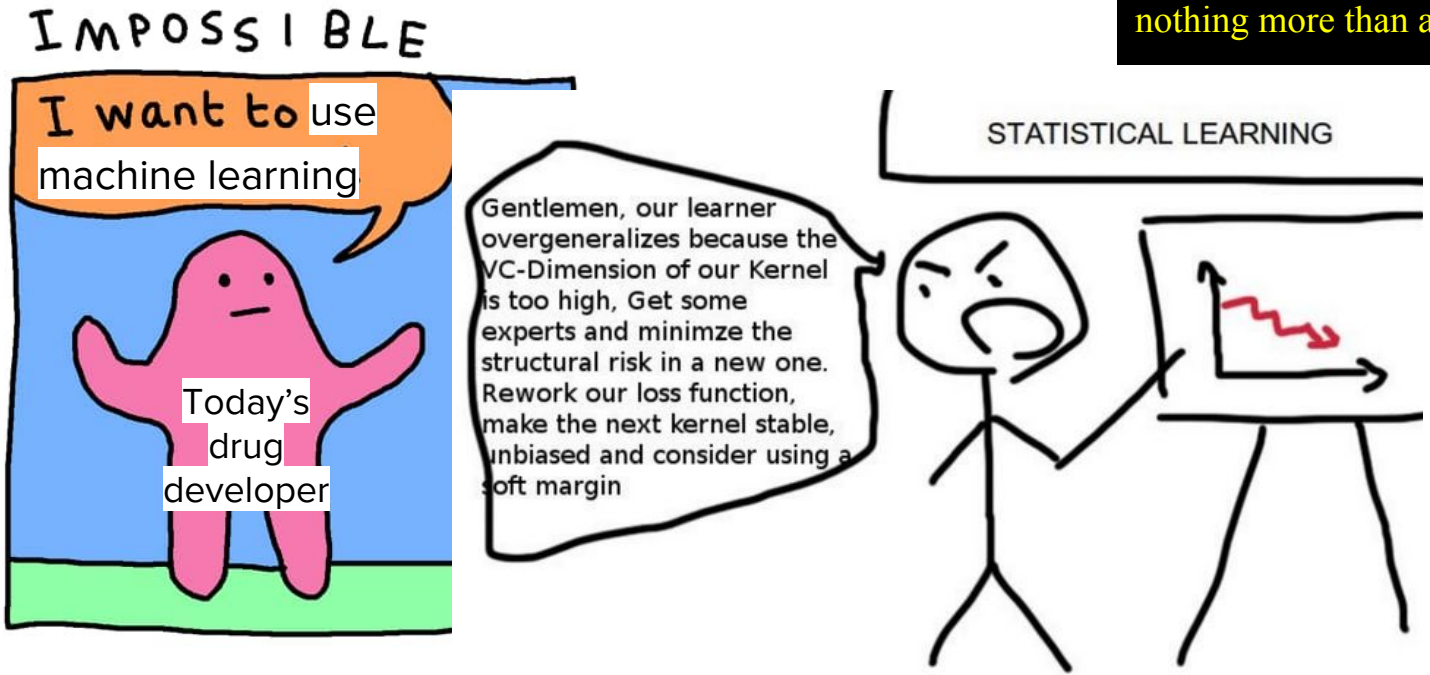
Drug discovery is hard and coming to CUP reinforces that fact :(

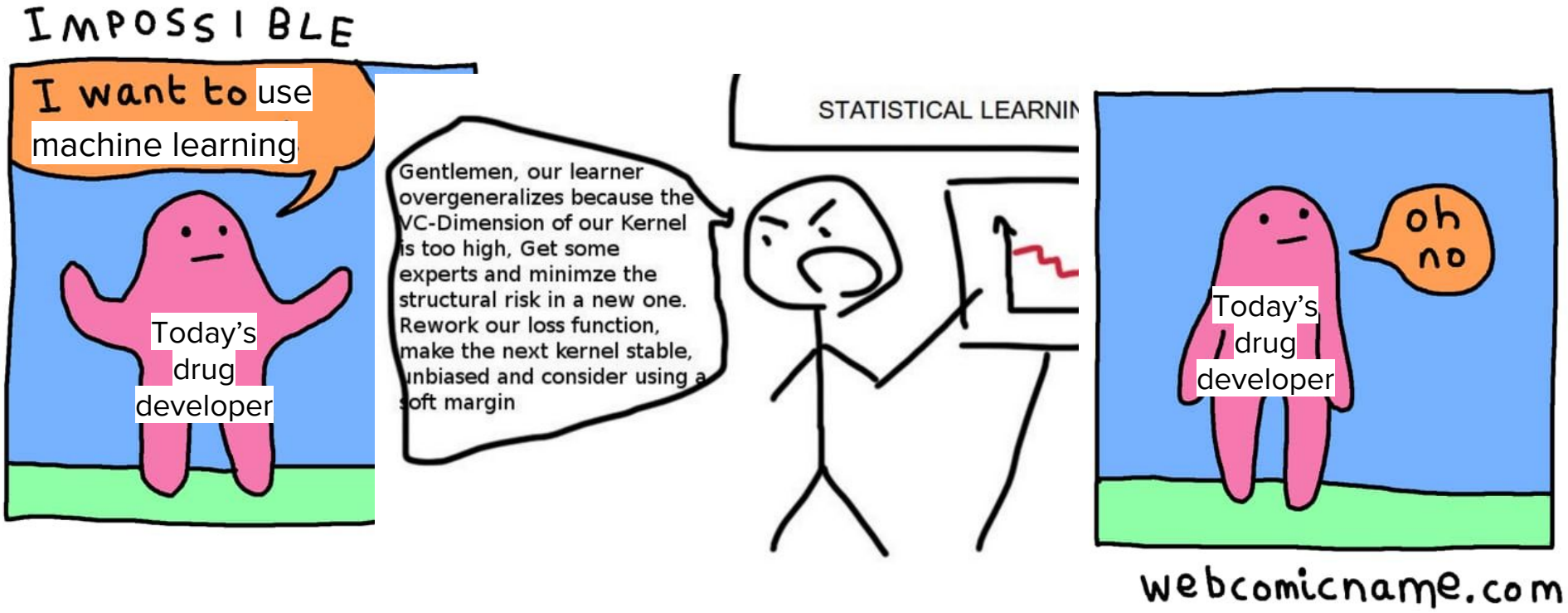


Because of the widely-reported successes in machine learning, people now hope that we might be able to shortcut some of that terrible hill climb by using these methods...

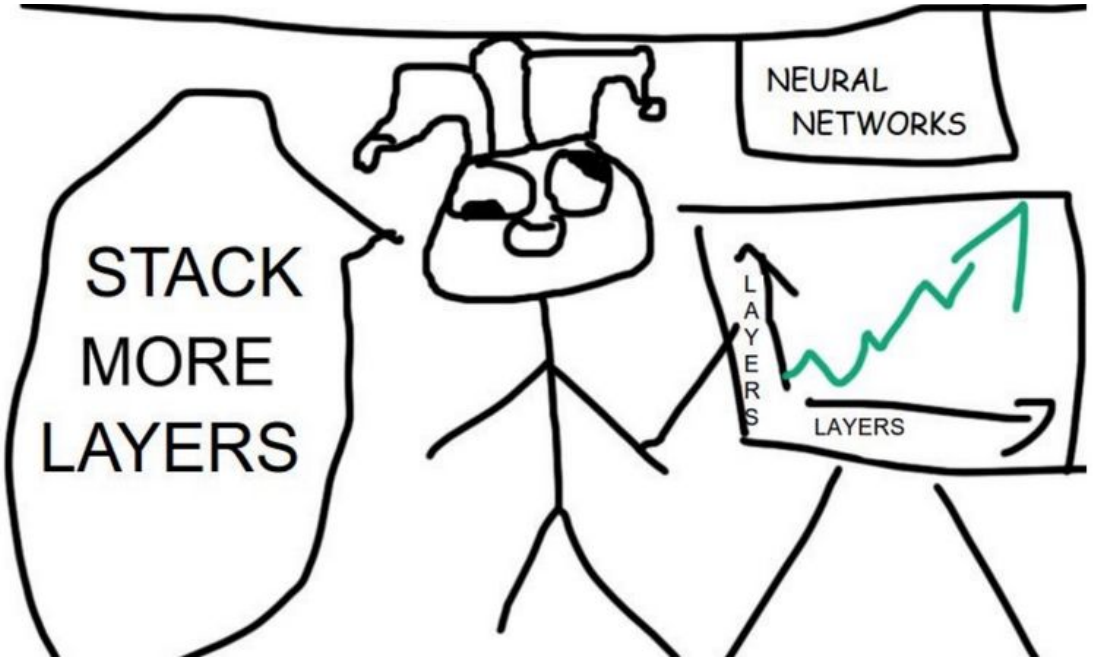


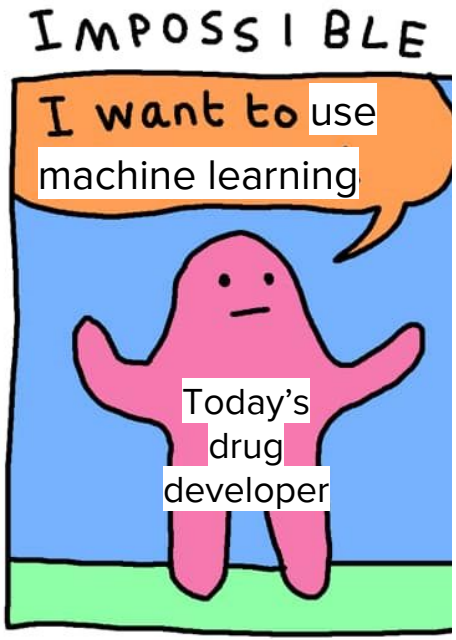
But we've been trying to use (traditional) ML in drug discovery for years, with limited success, and many rants at CUP explaining why we ought to do better statistics rather than going out on nothing more than a wing and a prayer.

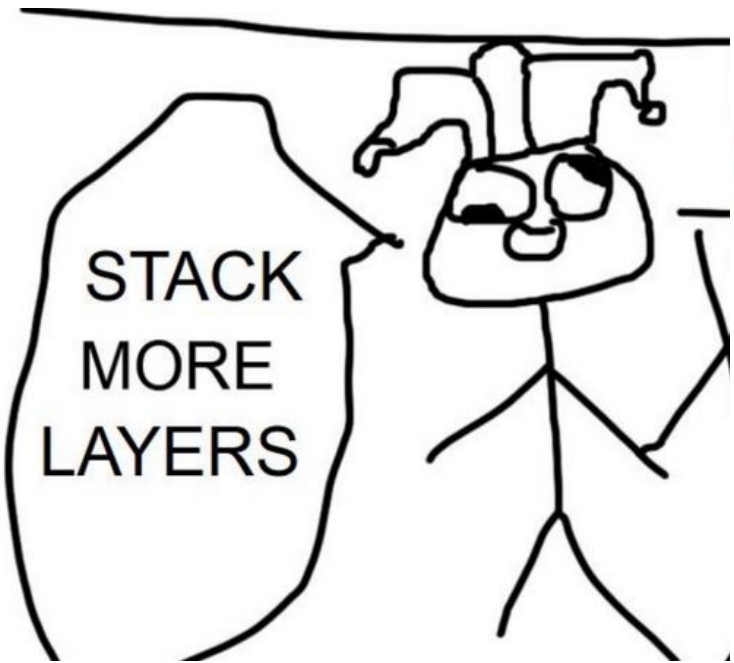




Of course, the comparative novelty of deep learning and neural networks makes us all hopeful yet again...







...not some show-and-tell about some method but

what principles we ought to be observing, or at least aware of...

Three Principles for ML

How come every problem is reducible to three points?

In the spirit of the prompt, this talk will discuss three principles that I propose can be used to decide whether a particular problem is likely to be a good candidate for a machine learning approach and how to design the surrounding data acquisition.

(Note that in this talk I largely follow current industry convention and use “AI” and “ML” synonymously; ML is more properly considered a statistical subset of a broader field of AI.)

The range of problems that have been proposed as targets for AI is vast, and the rate of ~~hype~~progress has motivated everyone to want a piece.

Will lay out a framework for thinking about the space, and demo with example problems:

- Discover lead compound
- Optimize compound solubility
- Discover compound efficacy biomarker

You might take issue with the particular characterizations of each demo problem; they're meant to be illustrative and approximate, not exactly correct for every scenario. Applying the principles to a particular project is left as an exercise for the reader (or for the author, should you choose to engage him as a consultant).

Question 1: Research Problems vs Business Problems

Three simple questions can help evaluate the difficulty of an AI/ML project:

1. **Has someone already gotten a computer to solve this problem (perhaps in another domain, or without ML)?**
2. **Do there exist humans who know how to solve this problem? (And are they on your team?)**
3. **Is a "good" solution well-defined?**

If the answers to #1 and #2 are "no", then this is a **research** problem ("can we train a computer to do X"). If #3 is also "no", it's a **really hard research problem**.

Question 1: Research Problems vs Business Problems

1. **Has someone already gotten a computer to solve this problem?**
    a. Yes: This is a business problem: will it work in this domain?
2. **Do there exist humans who know how to solve this problem?**
    a. Yes: This is an application research problem: can we get AI to replicate human performance?
3. **Is a “good” solution well-defined?**
    a. Yes: This is a method research problem: can we get a method to get a good result?
    b. No: Go back to the drawing board. This is too hard.
    c. (Aside: “good” is usually not as well-defined as you think it is.)

A useful keyword to look up to learn more about the last point (“good is not well-defined”) is “specification gaming”.

Demo matrix: Research vs Business

	Has a computer done it?	Has a human done it?	Is the objective well-defined?
Compound discovery	No	No	Yes-ish
Solubility Optimization	Maybe	Yes	Yes
Efficacy biomarker discovery	No	No	Maybe!

Question 2: How big is your data?

The Shape of Big Data

Note that most of the work in traditional cheminformatics around defining fingerprints, similarity metrics, etc. is “feature engineering”.

- Bytes come in different “shapes”.
- Having more **samples** is usually the more useful dimension for ML.
- Examples:
  - Ad clicks: **very tall** (few attributes, many events)
  - Facebook pictures: **tall** (low-res, lots of pictures)
  - Pathology slides: **wide** (many pictures, but ultra-high-res)
  - Genomics: **very wide** (1000s of samples, billions of attributes)
- Feature engineering is more important with fewer samples.

“wide dataset”

“tall dataset”

The example on the bottom right: noses are important for identifying faces, and have "local" structure: the pixels for a nose are all in a compact connected region. This assumption of locality often does not hold in chem / bio data sets.

The Structure of Big Data

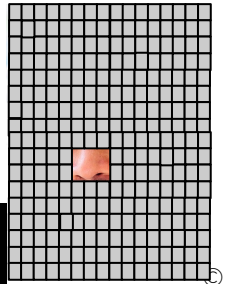
- Attributes in a dataset often have **structure**, or some relationship with each other; key target of feature engineering.
- The ability to exploit this structure is key for ML success, but often requires prior knowledge of the data: different models encode different structure (e.g., spatial structure in images).
- Leveraging biological structure is still a work in progress.

Data set 1: daily temperature in

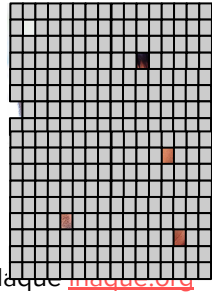
- San Jose
- Palo Alto
- Redwood City
- Daly City
- San Francisco

Data set 2: individual blood levels of

- C-reactive protein
- triglycerides
- insulin
- cortisol
- PSA



Local



Nonlocal

E.g., the assumption that we can break a compound's activity into that of its domains or functional groups is an example of assuming local structure - and that isn't always possible.

Question 2: Data: $ or $$$?

Will acquiring “tall” data in this domain be cheap (and big) or expensive (and small) (in materials, licenses, processing, time)?

Samples and data are still king; machine learning does not work without high-quality, high-volume input data with **many samples compared to #attributes**. Many strategies:

- Public data (but is it high-quality?)
- Licensing or partnership (is it high-volume and useful?)
- Internal “tweaks” on existing approaches
- Ground-up data acquisition

Question 2: Data - big/cheap or expensive/small?

- The core assumption of ML is that **enough** data is captured to reflect even rare events.
- Remember: it is not enough for your dataset to be big; it needs to be tall.
- The wider the dataset, the more informative, but usually the more expensive as well.
- Approaches squeezing more out of existing (e.g., unstructured) data can be valuable, even if manual. See e.g. Flatiron Health.

The Unreasonable Effectiveness of Data

*"A trillion word corpus...**captures even very rare aspects of human behavior**. So, this corpus could serve as the basis of a complete model for certain tasks - if only we knew how to extract the model from the data."*

Halevy et al. *IEEE Intel Sys* 2009

Demo matrix: Data size and expense

	Does a large data set exist?	**Cost of acquiring data**
Compound discovery	Sort of	Moderate; outsourceable
Solubility Optimization	Yes	Moderate; outsourceable
Efficacy biomarker discovery	No	Expensive

Question 3: Feedback - fast or slow?

1. **How long does it take to realize that a wrong answer is wrong?**
2. **How long to realize that a right answer is right?**
3. **What are the consequences to a wrong answer?**

In almost all applications, the prediction is the goal. Learning systems only improve if they can get feedback.

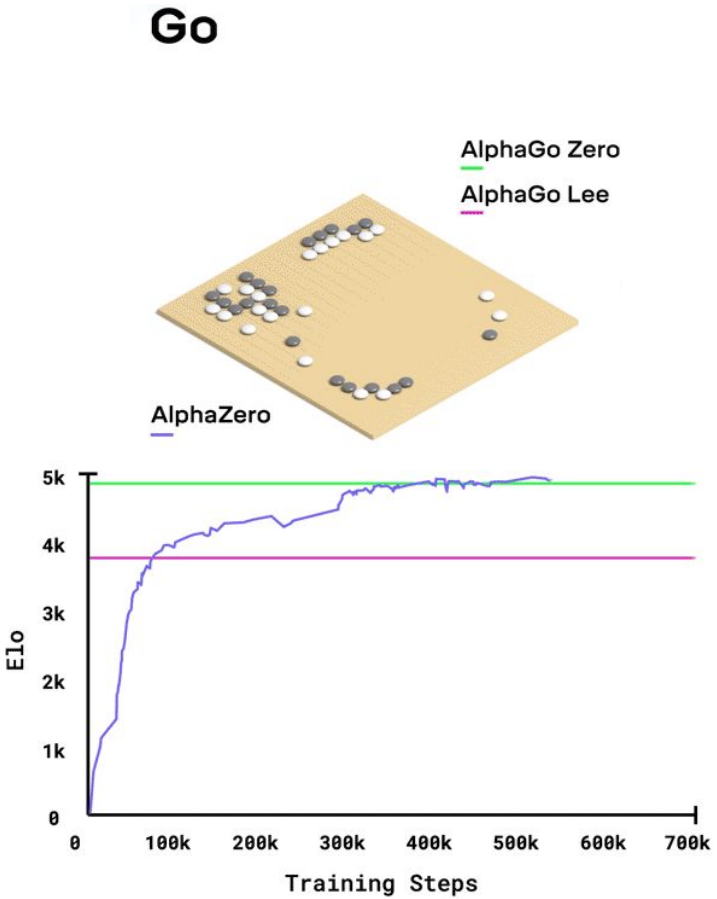
Especially important if answer to question 1 is “no existing computer system exists to solve this”.

Question 3: Rapid Feedback

Significant advances in machine learning performance have come in domains with rapid feedback on result quality:

- Image synthesis / recognition
- Games
- Robotics (via simulation)

Most biological problems have result latency measured in **years**, not microseconds.



DeepMind: shedding new light on the grand games of chess, shogi, and Go.

Demo matrix: Rapid feedback

	How quickly does feedback come?
Compound discovery	**Weeks** - in vitro studies **Years** - human studies
Solubility Optimization	**Minutes-Days** - if compound is at hand **Days-Weeks** - if commercially available **Weeks-Months** - if synthesis needed
Efficacy biomarker discovery	**Weeks-months** - if existing trial patients can be used **Years** - for a new trial

Where do we go from here?

The Drug Discovery Checklist

I wrote up a "checklist" to help evaluate new ideas in drug discovery. People found it amusing. You should check it out.

Your paper/post/pitch-deck advocates a

(X) machine learning ( ) genetics based ( ) patient stratification ( ) massively parallel experimental

approach to improving NDA success rates. Your idea will not work. Here is why it won't work. (One or more of the following may apply to your particular idea, and it may have other flaws which will vary from jurisdiction to jurisdiction depending on regulatory mood, population stratification, and differences in IP protection.)

https://ihaque.org/posts/2019/03/05/drug-discovery-checklist/

The Drug Discovery Checklist

"Tall" data is really, really challenging in chemistry because of just how superexponentially big chemistry really is. How can we try to improve the situation?

Your paper/post/pitch-deck advocates a

(X) machine learning ( ) genetics based ( ) patient stratification ( ) massively parallel experimental

approach to improving NDA success rates. Your idea will not work. Here is why it won't work. (One or more of the following may apply to your particular idea, and it may have other flaws which will vary from jurisdiction to jurisdiction depending on regulatory mood, population stratification, and differences in IP protection.)

(X) Chemical space is really big. You won't believe just how vastly, hugely, mind-bogglingly big it is. I mean, you may think there's a lot of variation in a phage library, but that's just peanuts to chemical space.

A spirit to embiggen the smallest dataset

- Genomics and proteomics data are typically sample constrained: too wide
- Chemical data is not only sample constrained, it is also feature constrained: simultaneously too wide and too short
  - ROCS Shape, ROCS Color, ZAP, etc. are all hand-engineered features designed to overcome this
  - GCNs et al. are automated ways to try to overcome this.

An interesting distinction between genomics / proteomics and chemistry is we typically have a LOT of measurements per sample in the former, but far fewer in the latter -- meaning that chemistry also tends to be too narrow. (Manually programmed) computed properties like shape and electrostatics are one way we've worked on this; people are trying methods like graph convolutional networks as possible ways to automatically infer new features.

Measure when you can, model if you must

- Competing exponentials: compute and NGS
- Two interesting sources of data for training models:

1. Data from simulations: we heard about this yesterday
2. Massively parallel assays (with MS/NGS readout)
    a. Huge dataset height (1e6-1e7 in one go)
    b. Potentially huge width (certainly in transcriptomics/proteomics x single-cell)

- IMO, massively parallel assays are the next big frontier for ML, but requires close design collaboration with the experimentalists.

A challenge with automatic feature discovery is having enough information (beyond atomic connectivity) to guide the learning. One route is to use physics-guided simulation to derive observables for training (and there was a whole session on this at CUP); another interesting one is the explosion in high-throughput experimental capabilities.

The big organizational challenge here is that the experiment needs to be co-designed with the analysis: on one hand, you need computational scientists who can think about the underlying assay and think about what can be queried, but also experimental scientists who work together with the computational scientists under the assumption that hand-analysis of the data will likely be opaque and impossible.

Measure when you can, model if you must

d NGS

aining models:

ut this yesterday

GS readout)

go)

anscriptomics/proteomics x single-cell)

e next big frontier for ML, but requires

xperimentalists.

Now, you might hear “learn from simulations” and think, “well, that’s going to be a ‘garbage-in-garbage-out’ situation”...

But the only thing that's worse is "real" high-throughput experimental data!

Measure when you can, model if you must

Yes, this scenario has been discussed in the checklist :)

Measure when you can, model if you must

d NG

(X) No one does good sequencing. No, you don't.

( ) Placebo is a pretty good standard of care.

(X) Most compounds in your screening library probably degraded or were mislabeled in the first place.

( ) Existing IP held by larger players who will sue you out of existence.

( ) Poor IP protection in the target market.

(X) Batch effects

e nex

xperi

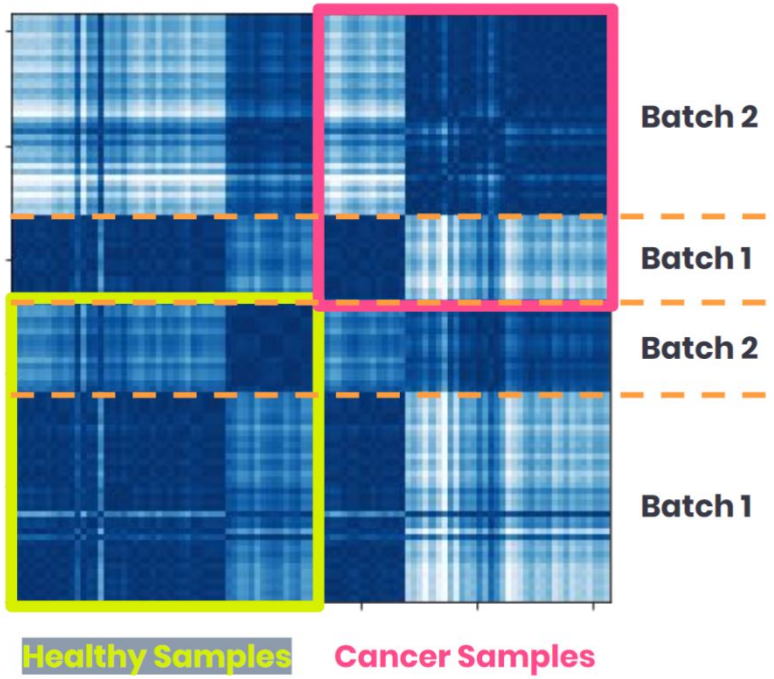
The example on the right is drawn from another talk I've given (linked below), looking at sequencing data from patients with and without cancer, sequenced in two batches. Samples sequenced on the same batch are more correlated with each other than samples with the same disease state.

Experiments: the worst, except for all other options

- All high-throughput data has terrible error bars (random error) as well as randomly-systematic (batch) error.

- Protip: get all the metadata you can; remove `KFold` from your vocabulary. Always stratify.

We wrote a paper on methods for confounder control (also on CRC detection): **doi:10.1101/478065**



https://ihaque.org/static/talks/20180703-deepchem.pdf

A topic discussed the previous day at CUP was ML models to improve the results from “low-accuracy” quantum chemistry simulations to bring them up to the quality of more-complicated models. Could something similar be done to correct systematic biases in high-throughput experimental data?

Experiments: the worst, except for all other options

- All high-throughput data has terrible error bars (random error) as well as randomly-systematic (batch) error.

- Protip: get all the metadata you can; remove `KFold` from your vocabulary. Always stratify.

We wrote a paper on methods for confounder control (also on CRC detection): **doi:10.1101/478065**

Musing: we’re building models to improve “crappy” MP2 and bring it up to CCSD(T).

Can we build models to improve “crappy” HTS data and make them more useful for downstream usage?

...or at least to model and incorporate their uncertainty?

Another checklist shoutout!

Conclusions

(✗) Ideas similar to yours are easy to come up with, yet none have ever been shown effective.

Conclusions

- The excitement around deep learning has been driven by a handful of domains with very particular characteristics:
  - Well-defined tasks which humans can largely (but slowly) perform
  - Huge amounts of instances with tractable (mostly local structure)
  - Rapid feedback on model quality
  - * also some nicely curated datasets

- Rather few of these apply to problems in chemical/biological learning. If your model works, be diligent about checking for confounders.
- Pairing massively parallel experiment with models that can clean up or harness their errors could be a neat way forward.

Hit me with questions: @imranshaque (Twitter) / ish@ihaque.org

(I'm also available to consult: consulting@ihaque.org)